HITACHI
Inspire the Next

Guidelines – Merging Custom Configuration Files for Pentaho Upgrades

HITACHI
Inspire the Next

Change log (if you want to use it):

Date	Version	Author	Changes

Contents

This page intentionally left blank.

Overview

This document is intended as a guide to assist you in upgrading to Pentaho 7.0, and from there to 8.0.

Some of the things discussed here include which files would need to be merged if you have a lot of customizations, notes on upgrading Pentaho plugins, and what to configure if you are using CAS security.

The intention of this document is to speak about topics generally; however, these are the specific versions covered here:

Software	Version(s)
Pentaho	if upgrading from 6.x to 7.0, and on to 8.0

The Components Reference in Pentaho Documentation has a complete list of supported software and hardware.

Merge Configuration Files

A few configuration files have changed between Pentaho 6.x and Pentaho 7.0. If you have customized any of the listed files in the Customized Files section, if you use plugins that depend on Spring Security, or if you are using Central Authentication Service (CAS), or Integrated Windows Authentication (IWA), you can find information related to these topics in these sections:

- Customized Files
- Spring Security Plugins
- CAS and IWA

*If you have **not customized** any of these configuration files listed in this document, you are ready to migrate your data.*

Customized Files

If you have customized any of these files, you will need to update your corresponding files to accommodate your customizations and successfully complete the upgrade.

We recommend you make backups of each of these files before altering them.

This table contains a list of the files that you might need to merge:

Table 1: Files to Merge

File Location	File Name
…\pentaho-solutions\system	`applicationContext-spring-security.xml`
	`applicationContext-spring-security-cas.xml`
	`applicationContext-spring-security-jackrabbit.xml`
	`applicationContext-spring-security-jdbc.xml`
	`applicationContext-spring-security-ldap.xml`
	`applicationContext-spring-security-memory.xml`
	`applicationContext-spring-security-superuser.xml`
	`pentahoObjects.spring.xml`
	`repository.spring.xml`
…\tomcat\webapps\pentaho\WEB-INF	`web.xml`

Spring Security Plugins

If your plugins are not dependent on Spring, then they will require no extra effort. However, if your plugins have dependencies on **Spring Security**, then you will need to merge those files as well.

CAS and IWA

If you are using **CAS** or **IWA**, you will need to apply a couple of specific updates.

- If you have customized any of the **files related to IWA**, please contact Support for assistance.
- If you want to use CAS with Pentaho 7.0, you will need to follow the instructions in [Switch to Central Authentication Service (CAS)](#) under SSO Security in Pentaho Documentation.

Class Mappings Reference Table

This is intended as a supplemental reference to assist in upgrading from Pentaho 6.0 to any higher version. Some of the things contained here include a class mappings reference table for Spring Security upgrades and compile error fixes reference tables for repository, extensions, and core compile errors.

The information in this reference table covers class mappings from Spring Security 2.0.x Classes to Spring Security 4.1.x Classes to help you through a Spring upgrade, if needed.

Spring Security 2.0.x Class	Spring Security 4.1.x Class
`org.springframework.security.context.SecurityContextHolder`	`org.springframework.security.core.context.SecurityContextHolder`
`org.springframework.security.Authentication`	`org.springframework.security.core.Authentication`
`org.springframework.security.util.FilterChainProxy`	`org.springframework.security.web.FilterChainProxy`
`org.springframework.security.providers.anonymous.AnonymousProcessingFilter`	`org.springframework.security.web.authentication.AnonymousAuthenticationFilter`
`org.springframework.security.ui.basicauth.BasicProcessingFilter`	`org.springframework.security.web.authentication.www.BasicAuthenticationFilter`
`org.springframework.security.ui.basicauth.BasicProcessingFilterEntryPoint`	`org.springframework.security.web.authentication.www.BasicAuthenticationEntryPoint`
`org.springframework.security.ui.ExceptionTranslationFilter`	`org.springframework.security.web.access.ExceptionTranslationFilter`
`org.springframework.security.ui.webapp.AuthenticationProcessingFilterEntryPoint`	`org.springframework.security.web.authentication.LoginUrlAuthenticationEntryPoint`
`org.springframework.security.ui.webapp.AuthenticationProcessingFilter`	`org.springframework.security.web.authentication.UsernamePasswordAuthenticationFilter`
`org.springframework.security.context.HttpSessionContextIntegrationFilter`	`org.springframework.security.web.context.HttpSessionSecurityContextRepository`
`org.springframework.security.vote.AffirmativeBased`	`org.springframework.security.access.vote.AffirmativeBased`
`org.springframework.security.vote.AuthenticatedVoter`	`org.springframework.security.access.vote.AuthenticatedVoter`

Spring Security 2.0.x Class	Spring Security 4.1.x Class
`org.springframework.security.intercept.web.FilterSecurityInterceptor`	`org.springframework.security.web.access.intercept.FilterSecurityInterceptor`
`org.springframework.security.acl.basic.AclObjectIdentity`	`org.springframework.security.acls.model.ObjectIdentity`
`org.springframework.security.GrantedAuthority`	`org.springframework.security.core.GrantedAuthority`
`org.springframework.security.userdetails.User`	`org.springframework.security.core.userdetails.User`
`org.springframework.security.core.AuthenticationServiceException`	`org.springframework.security.authentication.AuthenticationServiceException`
`org.springframework.security.AuthenticationServiceException`	`org.springframework.security.authentication.AuthenticationServiceException`
`org.springframework.security.AuthorizationServiceException`	`org.springframework.security.access.AuthorizationServiceException`
`org.springframework.security.providers.UsernamePasswordAuthenticationToken`	`org.springframework.security.authentication.UsernamePasswordAuthenticationToken`
`org.springframework.security.adaptersada.PrincipalSpringSecurityUserToken`	`org.springframework.security.authentication.UsernamePasswordAuthenticationToken`
`org.springframework.security.core.GrantedAuthorityImpl`	`org.springframework.security.core.authority.SimpleGrantedAuthority`
`org.springframework.security.vote.BasicAclEntryVoter`	`org.springframework.security.acls.AclEntryVoter`
`org.springframework.security.vote.AccessDecisionVoter`	`org.springframework.security.access.AccessDecisionVoter`
`org.springframework.security.acl.AclEntry`	`org.springframework.security.acls.model.Acl`
`org.springframework.security.ConfigAttributeDefinition`	`Collection<org.springframework.security.access.ConfigAttribute>`
`org.springframework.security.ConfigAttribute`	`org.springframework.security.access.ConfigAttribute`
`org.springframework.security.AuthorizationServiceException`	`org.springframework.security.access.AuthorizationServiceException`
`org.springframework.security.afterinvocation.AfterInvocationProvider`	`org.springframework.security.access.AfterInvocationProvider`
`org.springframework.security.AccessDeniedException`	`org.springframework.security.access.AccessDeniedException`
`org.springframework.security.acl.AclManager`	`org.springframework.security.acls.model.AclService`

Spring Security 2.0.x Class	Spring Security 4.1.x Class
`org.springframework.security.concurrent.SessionRegistry`	`org.springframework.security.core.session.SessionRegistry`
`org.springframework.security.concurrent.SessionInformation`	`org.springframework.security.core.session.SessionInformation`
`org.springframework.security.SecurityConfig`	`org.springframework.security.access.SecurityConfig`
`org.springframework.security.AuthorizationServiceException`	`org.springframework.security.access.AuthorizationServiceException`
`org.springframework.security.providers.encoding.PasswordEncoder`	`org.springframework.security.authentication.encoding.PasswordEncoder`
`org.springframework.security.ui.WebAuthenticationDetails`	`org.springframework.security.web.authentication.WebAuthenticationDetails`
`org.springframework.security.providers.dao.DaoAuthenticationProvider`	`org.springframework.security.authentication.dao.DaoAuthenticationProvider`
`org.springframework.security.ui.switchuser.SwitchUserGrantedAuthority`	`org.springframework.security.web.authentication.switchuser.SwitchUserGrantedAuthority`
`org.springframework.security.vote.AbstractAclVoter`	`org.springframework.security.access.vote.AbstractAclVoter`
`org.springframework.security.providers.anonymous.AnonymousAuthenticationToken`	`org.springframework.security.authentication.AnonymousAuthenticationToken`
`org.springframework.security.afterinvocation.AfterInvocationProvider`	`org.springframework.security.access.AfterInvocationProvider`
`org.springframework.security.AuthenticationManager`	`org.springframework.security.authentication.AuthenticationManager`
`org.springframework.security.ui.cas.CasProcessingFilter`	`org.springframework.security.cas.web.CasAuthenticationFilter`
`org.springframework.security.ui.rememberme.NullRememberMeServices`	`org.springframework.security.web.authentication.NullRememberMeServices`
`org.springframework.security.util.UrlUtils`	`org.springframework.security.web.util.UrlUtils`
`org.springframework.security.providers.AuthenticationProvider`	`org.springframework.security.authentication.AuthenticationProvider`
`org.springframework.security.userdetails.jdbc.JdbcDaoImpl`	`org.springframework.security.core.userdetails.jdbc.JdbcDaoImpl`

Spring Security 2.0.x Class	Spring Security 4.1.x Class
`org.springframework.security.BadCredentialsException`	`org.springframework.security.authentication.` `BadCredentialsException`
`org.springframework.security.userdetails.memory.` `UserAttribute`	`org.springframework.security.core.userdetails.memory.` `UserAttribute`
`org.springframework.security.context.SecurityContext`	`org.springframework.security.core.context.SecurityContext`
`org.springframework.security.providers.` `TestingAuthenticationToken`	`org.springframework.security.authentication.` `TestingAuthenticationToken`
`org.springframework.security.userdetails.ldap.` `LdapUserDetails`	`org.springframework.security.ldap.userdetails.` `LdapUserDetails`
`org.springframework.security.vote.RoleVoter`	`org.springframework.security.access.vote.RoleVoter`
`org.springframework.security.intercept.method.` `aopalliance.MethodSecurityInterceptor`	`org.springframework.security.access.intercept.` `aopalliance.MethodSecurityInterceptor`
`org.springframework.security.vote.UnanimousBased`	`org.springframework.security.access.vote.UnanimousBased`
`org.springframework.security.afterinvocation.` `AfterInvocationProviderManager`	`org.springframework.security.access.intercept.` `AfterInvocationProviderManager`
`org.springframework.security.providers.ProviderManager`	`org.springframework.security.authentication.` `ProviderManager`
`org.springframework.security.ui.` `AuthenticationDetailsSourceImpl`	`org.springframework.security.authentication.` `AuthenticationDetailsSourceImpl`
`org.springframework.security.afterinvocation.` `BasicAclEntryAfterInvocationCollectionFilt`	`org.springframework.security.acls.afterinvocation.` `AclEntryAfterInvocationCollectionFiltering`
`org.springframework.security.userdetails.memory.` `InMemoryDaoImpl`	`org.springframework.security.provisioning.` `InMemoryUserDetailsManager`
`org.springframework.security.event.authentication.` `LoggerListener`	`org.springframework.security.authentication.event.` `LoggerListener`
`org.springframework.security.wrapper.` `SecurityContextHolderAwareRequestFilter`	`org.springframework.security.web.servletapi.` `SecurityContextHolderAwareRequestFilter`
`org.springframework.security.afterinvocation.` `BasicAclEntryAfterInvocationProvider`	`org.springframework.security.acls.afterinvocation.` `AclEntryAfterInvocationProvider`
`org.springframework.security.concurrent.` `SessionRegistryImpl`	`org.springframework.security.core.session.` `SessionRegistryImpl`

Spring Security 2.0.x Class	Spring Security 4.1.x Class
`org.springframework.security.userdetails.` `UserDetailsService`	`org.springframework.security.core.userdetails.` `UserDetailsService`
`org.springframework.security.userdetails.` `UsernameNotFoundException`	`org.springframework.security.core.userdetails.` `UsernameNotFoundException`
`org.springframework.security.userdetails.UserDetails`	`org.springframework.security.core.userdetails.UserDetails`
`org.springframework.security.context.` `SecurityContextHolderStrategy`	`org.springframework.security.core.context.` `SecurityContextHolderStrategy`
`org.springframework.security.context.` `SecurityContextImpl`	`org.springframework.security.core.context.` `SecurityContextImpl`
`org.springframework.security.event.authentication.` `AuthenticationSuccessEvent`	`org.springframework.security.authentication.event.` `AuthenticationSuccessEvent`
`org.springframework.security.ldap.LdapAuthoritiesPopula` `tor`	`org.springframework.security.ldap.userdetails.` `LdapAuthoritiesPopulator`
`org.springframework.security.userdetails.ldap.` `LdapUserDetailsMapper`	`org.springframework.security.ldap.userdetails.` `LdapUserDetailsMapper`
`org.springframework.security.userdetails.memory.UserMap`	`org.springframework.security.core.userdetails.memory.` `UserMap`
`org.springframework.security.ldap.populator.` `DefaultLdapAuthoritiesPopulator`	`org.springframework.security.ldap.userdetails.` `DefaultLdapAuthoritiesPopulator`
`org.springframework.security.userdetails.ldap.` `LdapUserDetailsService`	`org.springframework.security.ldap.userdetails.` `LdapUserDetailsService`
`org.springframework.security.ldap.LdapUserSearch`	`org.springframework.security.ldap.search.LdapUserSearch`
`org.springframework.security.AuthenticationException`	`org.springframework.security.core.AuthenticationException`
`org.springframework.security.providers.ldap.` `LdapAuthenticationProvider`	`org.springframework.security.ldap.authentication.` `LdapAuthenticationProvider`
`org.springframework.security.providers.ldap.` `LdapAuthenticator`	`org.springframework.security.ldap.authentication.` `LdapAuthenticator`
`org.springframework.security.userdetails.memory.` `UserMapEditor`	`org.springframework.security.core.userdetails.memory.` `UserMapEditor`
`org.springframework.security.providers.dao.UserCache`	`org.springframework.security.core.userdetails.UserCache`

Spring Security 2.0.x Class	Spring Security 4.1.x Class
`org.springframework.security.providers.dao.` `NullUserCache`	`org.springframework.security.core.userdetails.cache.` `NullUserCache`
`org.springframework.security.providers.dao.cache.` `NullUserCache`	`org.springframework.security.core.userdetails.cache.` `NullUserCache`
`org.springframework.security.userdetails.memory.` `UserAttributeEditor`	`org.springframework.security.core.userdetails.memory.` `UserAttributeEditor`
`org.springframework.security.ui.logout.LogoutHandler`	`org.springframework.security.web.authentication.logout.` `LogoutHandler`
`org.springframework.security.ui.` `AuthenticationEntryPoint`	`org.springframework.security.web.AuthenticationEntryPoint`

`org.springframework.security.ui.preauth.j2ee.`
`J2eePreAuthenticatedProcessingFilter`
	`org.springframework.security.web.authentication.preauth.`
	`j2ee.`
	`J2eePreAuthenticatedProces singFilter`

`org.springframework.security.ui.preauth.` `PreAuthenticatedProcessingFilterEntryPoint`	`org.springframework.security.web.authentication.` `Http403ForbiddenEntryPoint`
`org.springframework.security.vote.UnanimousBased`	`org.springframework.security.access.vote.UnanimousBased`
`org.springframework.security.providers.encoding.` `PlaintextPasswordEncoder`	`org.springframework.security.authentication.encoding.` `PlaintextPasswordEncoder`
`org.springframework.security.providers.anonymous.` `AnonymousAuthenticationProvider`	`org.springframework.security.authentication.` `AnonymousAuthenticationProvider`
`org.springframework.security.ui.logout.LogoutFilter`	`org.springframework.security.web.authentication.logout.` `LogoutFilter`
`org.springframework.security.ui.logout.` `SecurityContextLogoutHandler`	`org.springframework.security.web.authentication.logout.` `SecurityContextLogoutHandler`
`org.springframework.security.wrapper.` `SecurityContextHolderAwareRequestWrapper`	`org.springframework.security.web.servletapi.` `SecurityContextHolderAwareRequestWrapper`
`org.springframework.security.ui.AccessDeniedHandlerImpl`	`org.springframework.security.web.access.` `AccessDeniedHandlerImpl`
`org.springframework.security.providers.ldap.` `authenticator.BindAuthenticator`	`org.springframework.security.ldap.authentication.` `BindAuthenticator`
`org.springframework.security.util.FilterToBeanProxy`	`org.springframework.web.filter.DelegatingFilterProxy`

Spring Security 2.0.x Class	Spring Security 4.1.x Class
`org.springframework.security.ui.savedrequest.SavedRequest`	`org.springframework.security.web.savedrequest.SavedRequest`
`org.springframework.security.providers.encoding.Md5PasswordEncoder`	`org.springframework.security.authentication.encoding.Md5PasswordEncoder`
`org.springframework.security.ui.cas.CasProcessingFilterEntryPoint`	`org.springframework.security.cas.web.CasAuthenticationEntryPoint`
`org.springframework.security.userdetails.ldap.UserDetailsContextMapper`	`org.springframework.security.ldap.userdetails.UserDetailsContextMapper`
`org.springframework.security.providers.dao.cache.EhCacheBasedUserCache`	`org.springframework.security.core.userdetails.cache.EhCacheBasedUserCache`
`org.springframework.extensions.jcr.support.ListSessionHolderProviderManager`	`org.pentaho.platform.repository2.unified.jcr.sejcr.JackrabbitListSessionHolderProviderManager`
`org.springframework.security.acl.AclEntry`	`org.pentaho.platform.api.engine.IAclEntry`
`org.springframework.security.acl.basic.BasicAclEntry`	`org.pentaho.platform.api.engine.IPentahoBasicAclEntry`
`org.springframework.security.acl.basic.AbstractBasicAclEntry`	`org.pentaho.platform.engine.security.acls.PentahoAbstractBasicAclEntry`
`org.springframework.security.acl.basic.GrantedAuthorityEffectiveAclsResolver`	`org.pentaho.platform.engine.security.acls.PentahoGrantedAuthorityEffectiveAclsResolver`

Compile Error Fixes References

Here is a list of common compilation error messages that you may receive, as well as where to find them and changes needed to fix them.

- [Repository – Compile Error Fixes](#)
- [Extensions – Compile Error Fixes](#)
- [Extensions – Test Compile Error Fixes](#)
- [Core – Compile Error Fixes](#)
- [Core – Test Compile Error Fixes](#)

Repository – Compile Error Fixes

Class	Code lines	Change from	Change to
`DefaultUnifiedRepositoryBase`	307, 308	`authorities`	`Arrays.asList( authorities )`
`DefaultUnifiedRepositoryBase`	324, 326	`repositoryAdminAuthorities`	`Arrays.asList( repositoryAdminAuthorities )`
`MockUnifiedRepository`	1315	`GrantedAuthority[]`	`Collection<? extends GrantedAuthority>`
`BackingRepositoryLifecycleManager AuthenticationSuccessListenerTest`	191	`GrantedAuthority[]`	`Collection<? extends GrantedAuthority>`
`UserRoleDaoUserDetailsServiceTest`	91 to 108	`userFromService.getAuthorities ()[X].getAuthority()`	`userFromService.getAuthorities().toArray()[X]`

Extensions – Compile Error Fixes

Class	Code lines	Change from	Change to
NoOpLdapAuthorities Populator	41	GrantedAuthority[]	Collection<? extends GrantedAuthority>
PentahoBasicProcessing Filter	60	public void doFilterHttp(HttpServletRequest request, HttpServletResponse response, FilterChain chain) throws IOException, ServletException {	public void doFilter(ServletRequest servletRequest, ServletResponse servletResponse, FilterChain chain) throws IOException, ServletException { HttpServletRequest request = (HttpServletRequest) servletRequest; HttpServletResponse response = (HttpServletResponse) servletResponse;
PentahoBasicProcessing Filter	121	super.doFilterHttp	super.doFilter
RequestParameter AuthenticationFilter	158	response	(HttpServletResponse) response
RequestParameter FilterEntryPoint	50	commence(final ServletRequest request, final ServletResponse response	commence(final HttpServletRequest request, final HttpServletResponse response
MondrianAbstract PlatformUserRoleMapper	156	authentication. getAuthorities()	authentication.getAuthorities().toArray (new GrantedAuthority[]{})
NotFoundUserDetails	39	GrantedAuthority[]	Collection<? extends GrantedAuthority>
JdbcUserRoleList Service	235	user.getAuthorities(). length	user.getAuthorities().toArray().length
DefaultLdap AuthenticationProvider	61	protected GrantedAuthority[] loadUserAuthorities	protected Collection<? extends GrantedAuthority> loadUserAuthorities
DefaultLdap AuthenticationProvider	62	super.loadUserAuthorities(userData, username, password)	super.loadUserAuthorities(userData, username, password).toArray(new GrantedAuthority[]{})

Class	Code lines	Change from	Change to
DefaultLdap Authentication Provider	71	`return authorities`	`return Arrays.asList( authorities )`
UnionizingLdapAuthorities Populator	52	```public GrantedAuthority[] getGrantedAuthorities(final DirContextOperations userData, final String username) { Set<GrantedAuthority> allAuthorities = new HashSet<GrantedAuthority>(); for (LdapAuthoritiesPopulator populator : populators) { GrantedAuthority[] auths = populator.getGrantedAuthorities(userData, username); if ((null != auths) && (auths.length > 0)) { allAuthorities.addAll(Arrays.asList(auths)); } } return allAuthorities.toArray(new GrantedAuthority[0]); }```	```public Collection<? extends GrantedAuthority> getGrantedAuthorities(final DirContextOperations userData, final String username) { Set<GrantedAuthority> allAuthorities = new HashSet<GrantedAuthority>(); for (LdapAuthoritiesPopulator populator : populators) { Collection<? extends GrantedAuthority> auths = populator.getGrantedAuthorities(userData, username); if ((null != auths) && (auths.size() > 0)) { allAuthorities.addAll(auths); } } return Arrays.asList(allAuthorities.toArray(new GrantedAuthority[]{})); }```
InMemoryUserRoleListService	105	```Arrays.asList(user.getAuthorities())```	```Arrays.asList(user.getAuthorities().toArray(new GrantedAuthority[]{}));```
DefaultLdapUserRoleListService	153	```Arrays.asList(user.getAuthorities())```	```Arrays.asList(user.getAuthorities().toArray(new GrantedAuthority[]{}));```
UserRoleListEnhancedUserMap	66	`user.getAuthorities()`	```user.getAuthorities().toArray(new GrantedAuthority[]{});```
SpringSecurityLoginModule	166	`authentication.getAuthorities()`	```authentication.getAuthorities().toArray(new GrantedAuthority[]{})```

Class	Code lines	Change from	Change to
ProxyTrustingFilter	225	HttpSessionContextIntegrationFilter.SPRING_SECURITY_CONTEXT_KEY	HttpSessionSecurityContextRepository.SPRING_SECURITY_CONTEXT_KEY
SystemRefreshResource	(multiple)	SecurityHelper.getInstance().isPentahoAdministrator(pentahoSession)	canAdminister()
WebTemplateHelper	99	SecurityHelper.getInstance().isPentahoAdministrator(pentahoSession)	canAdminister()
UserConsoleService	23	SecurityHelper.getInstance().isPentahoAdministrator(pentahoSession)	SystemUtils.canAdminister()
SchedulerService	294	SecurityHelper.getInstance().isPentahoAdministrator(pentahoSession)	canAdminister(getSession())

Extensions – Test Compile Error Fixes

Class	Code lines	Change from	Change to
AbstractSpringSecurityAuth	165, 227	providerManager.doAuthentication(auth)	providerManager.authenticate(auth)
AbstractSpringSecurityAuth	288	GrantedAuthority[]	Collection<? extends GrantedAuthority>
AbstractSpringSecurityAuth	288	authorities	Arrays.asList(authorities)
FileResourceTest	741, 743, 778, 804	new User(repositoryAdminUsername, password, true, true, true, true, repositoryAdminAuthorities)	new User(repositoryAdminUsername, password, true, true, true, true, Arrays.asList(repositoryAdminAuthorities))

Class	Code lines	Change from	Change to
FileResourceTest	760, 786	new UsernamePasswordAuthenticationToken (userDetails, password, authorities)	new UsernamePasswordAuthenticationToken(userDetails, password, Arrays.asList (authorities))
MetadataRepository LifecycleManagerTest	337, 375, 401	new User(repositoryAdminUsername, password, true, true, true, true, repositoryAdminAuthorities)	new User(repositoryAdminUsername, password, true, true, true, true, Arrays.asList(repositoryAdminAuthorities))
UserRoleDaoTest	341, 367	new User(repositoryAdminUsername, password, true, true, true, true, repositoryAdminAuthorities)	new User(repositoryAdminUsername, password, true, true, true, true, Arrays.asList(repositoryAdminAuthorities))
UserRoleDaoTest	343, 369	new UsernamePasswordAuthenticationToken (userDetails, password, authorities)	new UsernamePasswordAuthenticationToken(userDetails, password, Arrays.asList (authorities))
UserRoleDaoEncodeTest	341, 369	new User(repositoryAdminUsername, password, true, true, true, true, repositoryAdminAuthorities)	new User(repositoryAdminUsername, password, true, true, true, true, Arrays.asList(repositoryAdminAuthorities))
JdbcUserRoleList ServiceTests	241	new User(repositoryAdminUsername, password, true, true, true, true, repositoryAdminAuthorities)	new User(repositoryAdminUsername, password, true, true, true, true, Arrays.asList(repositoryAdminAuthorities))
JdbcUserRoleListService Tests	243	new UsernamePasswordAuthenticationToken (userDetails, password, authorities)	new UsernamePasswordAuthenticationToken(userDetails, password, Arrays.asList (authorities))
UserRoleMapperTest	136	auth.getAuthorities()	auth.getAuthorities().toArray(new GrantedAuthority[]{})

Core – Compile Error Fixes

Class	Code lines	Change from	Change to
`StandaloneSpringPentaho ObjectFactory`	77	(this is an addition; there is no code to replace)	`appCtx.refresh();`
`PentahoSystem`	86	`private static final String securityContextHolderStrategy = "org.pentaho.platform.engine.security .PentahoSecurityContextHolderStrategy";`	`private static final String securityContextHolderStrategy = SecurityContextHolder.MODE_INHERITABLE THREADLOCAL;`
`PentahoAuthentication SuccessListener`	69	`if ( event instanceof AuthenticationSuccessEvent ) {`	`if ( event instanceof AuthenticationSuccessEvent || event instanceof InteractiveAuthenticationSuccessEvent ) {`

Core – Test Compile Error Fixes

Class	Code lines	Change from	Change to
PentahoSubstring RoleVoter	110	`public boolean voteGenericTest(GrantedAuthority[] authorities, String[] strConfArr, int expectedResult) { Authentication authentication = new PrincipalSpringSecurityUserToken(StringUtils.EMPTY, StringUtils.EMPTY, StringUtils.EMPTY, authorities, null); ConfigAttributeDefinition config = new ConfigAttributeDefinition(strConfArr); return expectedResult == pentahoSubstringRoleVoter.vote(authentication, new Object(), config); }`	`public boolean voteGenericTest(GrantedAuthority[] authorities, String[] strConfArr, int expectedResult) { Authentication authentication = new UsernamePasswordAuthenticationToken(StringUtils.EMPTY, StringUtils.EMPTY, Arrays.asList(authorities)); Collection<ConfigAttribute> configAttributes = new ArrayList<ConfigAttribute>(); for(final String strConf : strConfArr){ configAttributes.add(new ConfigAttribute() { @Override public String getAttribute() { return strConf; } }); } return expectedResult == pentahoSubstringRoleVoter.vote(authentication, new Object(), configAttributes); }`
SecurityHelper Test	91	`GrantedAuthority[]`	`Collection<? extends GrantedAuthority>`
StandaloneSpring PentahoObject FactoryTest	234	(this is an addition; there is no code to replace)	`appCtx.refresh();`